Parallel Information Algorithm with Local Tuning for Solving Multidimensional GO Problems

YAROSLAV D. SERGEYEV   yaro@si.deis.unical.it
Institute of Systems Analysis and Information Technology of National Research Council, c/o DEIS, University of Calabria, 87036 Rende (CS), Italy and Software Department, University of Nizhni Novgorod, Gagarin Av. 23, Nizhni Novgorod, Russia

Abstract. In this paper we propose a new parallel algorithm for solving global optimization (GO) multidimensional problems. The method unifies two powerful approaches for accelerating the search : parallel computations and local tuning on the behavior of the objective function. We establish convergence conditions for the algorithm and theoretically show that the usage of local information during the global search permits to accelerate solving the problem significantly. Results of numerical experiments executed with 100 test functions are also reported.

Keywords: Global optimization, parallel computations, local tuning.

1. Introduction

Let us consider the problem of finding the global minimum and global minimizers of a function $\phi(y), y \in D$, where

$$D = \{x : a_i \le x_i \le b_i, 1 \le i \le N\}$$

is a hyperinterval in R^N. This problem is actively studied by many authors (see e.g. [6], [7], [8], [9], [16]). One of the promising approaches to attack this problem is in using the Peano type space-filling curves.

It is known (see [2], [16], [17]) that this multidimensional problem can be reduced to a one-dimensional one by using the curves. In this case we obtain

$$\min\{\phi(y) : y \in D\} = \min\{\phi(y(x)) : x \in [0,1]\}, \tag{1}$$

moreover, if the multidimensional function $\phi(y)$ satisfies the Lipschitz condition with the constant L over D then $\phi(y(x))$ over the interval $[0,1]$ satisfies the Hőlder condition

$$\mid \phi(y(x')) - \phi(y(x'')) \mid \le H \mid x' - x'' \mid^{1/N}, \quad x', x'' \in [0,1], \tag{2}$$

where H is the Hőlder constant and

$$H = 4Ld\sqrt{N}, \qquad d = \max\{b_i - a_i : 1 \le i \le N\}.$$

The constant H corresponds to the space-filling curve $y(x)$ which computational scheme is presented in [16] (see also [18]). A special procedure for fast calculating

the image $y(x) \in D$ on the curve for every given $x \in [0,1]$ has been proposed in these papers. The procedure has a high speed because, instead of constructing an approximation of the whole curve, it directly computes the coordinates of $y(x)$ in D for every given x. In addition, its parallel version has been introduced in [18].

For solving the problem (1), (2) the *information algorithm* (IA) has been proposed in [16], [17] for the case when the Lipschitz constant (and, therefore the Hőlder one) is unknown. A special procedure for estimating H on the base of information obtained during the search has been elaborated.

The power of parallel computations is widely used in global optimization to accelerate the search (different approaches are presented in [1], [5], [8] – [10], [14], [15], [18] – [20]). It has been shown that very often simple parallelizing ideas (parallel grid method or subdividing the search region in p subregions where p processors work in parallel) can lead to appearence of redundant function evaluations in comparison with an efficient sequential method. That is why one of the reasonable ways for usage of parallel computers consists in parallelizing fast sequential methods.

Wide numerical experiments and a deep theoretical study have shown (see [4], [16]) that IA has good speed characteristics in comparison with other methods which don't use derivatives. In [18] a *parallel information algorithm* (PIA) extending IA to the case of parallel computations has been introduced. Convergence conditions and estimates of speed up obtained in comparison with the original sequential IA have been established. A special study has been executed to obtain conditions which guarantee absence of redundant (in comparison with the sequential case) evaluations of $\phi(y(x))$. This idea has demonstrated to be fruitful and it was applied to parallelize some other sequential GO methods (see [3], [14], [15]). By generalizing this approach a class of parallel characteristical GO algorithms has been introduced and theoretically investigated in [5].

An alternative approach for acceleration of the global search consists in the following. It has been shown for different one-dimensional GO algorithms (see [11] – [13]) that using estimates of the Lipschitz constant L slows down the search for that subregions where the local Lipschitz constants are significantly less than L (hereinafter we shall call L the *global* Lipschitz constant). The sequential *information* GO *algorithm with local tuning* (IALT) has been introduced in [12] for solving the problem (1), (2). In that paper a special procedure has been proposed to estimate adaptively local constants in different areas of the search region to accelerate the search. Theoretical results and numerical experiments have shown that this approach permits to accelerate the search significantly in comparison with the original IA using esimates of the global Lipschitz constant.

In this paper we propose to unify both approaches and construct a *parallel* information algorithm with *local tuning* (PLT). In order to accelerate the search during every iteration the new method makes the following :

- it adaptively estimates *local Hőlder constants* in different subintervals of $[0,1]$ to tune itself on the local behaviour of the reduced objective function $\phi(y(x)), x \in [0,1]$;

- it simultaneously evaluates the objective function $\phi(y(x))$ at p different points in p intervals (having the highest probability to contain the global optimum) on p parallel computers;

- local information about $\phi(y(x))$ is used over the *whole* search region *during* the global search in contrast with traditional approaches (see e.g. [6], [7]) usually starting a local search in a neighbourhood of a global minimizer after stopping the global procedure.

It will be shown further that adaptive estimates of the local Hőlder constants introduced here permit to accelerate the search significantly in comparison with the parallel methods (see [5], [18]) using the global Lipschitz (Hőlder) constants or their estimates.

The rest of the paper is structured as follows. Section 2 describes the computational scheme of PLT and presents convergence results of the new type for the introduced parallel method. Section 3 contains results of a numerical comparison between IA, PIA, IALT and PLT executed with 100 two-dimensional functions usually used in literature for numerical comparison between the information GO methods (see [4], [5], [16], [18]). Some conclusions are presented in the last Section.

2.   The algorithm and its convergence conditions

To describe the algorithm we need some designations and definitions. We introduce a simplifying designation

$$f(x) = \phi(y(x)), \quad x \in [0, 1],$$

for the objective function $\phi(y(x))$. We call *trial* the operation of evaluating $f(x)$ at a point x. The character l we use as the counter of parallel iterations of PLT. In the course of every parallel iteration l we evaluate $f(x)$ at $p(l) > 1$ points using $p(l)$ parallel processors. We use the Peano curve approximation with the depth m presented in [16], [18] and a preset accuracy ϵ concorded with m by the inequality

$$\epsilon \geq 2^{-m}/(4\sqrt{N}).$$

Let us now present the computational scheme of the new method.

Step 0. Execute $q(l) > 1$ initial trials at the points $x^1 = 0, x^2 = 1$ and some internal points $x^3, x^4, \ldots, x^{q(l)}$ belonging to $(0, 1)$. Set $l = 1$.

Step 1. Points $x^1, x^2, \ldots, x^{q(l)}$ of the previous trials reorder by increasing of their coordinates using subscripts, i.e.

$$0 = x_1 < x_2 < \ldots < x_{q-1} < x_q = 1$$

where $q = q(l)$.

Step 2. Calculate the values μ_j estimating local Hőlder constants for the intervals $[x_{j-1}, x_j], 2 \leq j \leq q$, following the rules :

$$\mu_j = \max\{\lambda_j, \gamma_j, \xi\}, \ 2 \leq j \leq q, \tag{3}$$

where $\xi > 0$ is a small number, the value λ_j spying on the local information is determened by the formulae

$$\lambda_j = \max\{\frac{\mid f(x_i) - f(x_{i-1}) \mid}{\mid x_i - x_{i-1} \mid^{1/N}} : i \in I_j\}, \tag{4}$$

$$I_j = \left\{ \begin{array}{ll} \{2, 3\} & \text{if } j = 2 \\ \{j - 1, j, j + 1\} & \text{if } 3 \leq j \leq q - 1 \\ \{q - 1, q\} & \text{if } j = q \end{array} \right\}.$$

The last part of formulae (3) controls the global information accumulated during all the previous iterations and is represented by the value

$$\gamma_j = \mu(x_j - x_{j-1})^{1/N} / (X^{\max})^{1/N}, \tag{5}$$

where

$$\mu = \max\{\mid f(x_j) - f(x_{j-1}) \mid / (x_j - x_{j-1})^{1/N} : 2 \leq j \leq q\},$$

$$X^{\max} = \max\{x_i - x_{i-1} : 2 \leq i \leq q\}.$$

Step 3. For the intervals $[x_{j-1}, x_j], 2 \leq j \leq q$, compute their characteristics $R(j)$ as follows

$$R(j) = r\mu_j(x_j - x_{j-1})^{1/N} + \frac{(z_j - z_{j-1})^2}{r\mu_j(x_j - x_{j-1})^{1/N}} - (z_j + z_{j-1}), \quad 2 \leq j \leq q, \tag{6}$$

where $z_j = f(x_j), 1 \leq j \leq q$, and $r > 1$ is the reliability parameter of the method.

Step 4. Calculate the new trial points $x^{q+j} \in (0, 1), 1 \leq j \leq p$, and their images $y(x^{q+j}) \in D, 1 \leq j \leq p$, on the Peano curve m-approximation

$$x^{q+j} = 0.5(x_{t_j-1} + x_{t_j}) - (2r)^{-1}(\mid z_{t_j} - z_{t_j-1} \mid / \mu_{t_j})^N \text{sign}(z_{t_j} - z_{t_j-1}), \tag{7}$$

where

$$t_1 = \operatorname{argmax}\{R(i) : 1 < i \le q\},$$

$$t_j = \operatorname{argmax}\{R(i) : 1 < i \le q, i \ne t_s, 1 \le s \le j-1\}, 1 < j \le p,$$

$$p = p(l+1) \le q(l) - 1.$$

Step 5. Execute trials of the $(l+1)$th iteration at the points $y(x^{q+j}), 1 \le j \le p$, where $x^{q+j}, 1 \le j \le p$, are from (7).

Step 6. If the stopping rule

$$\min\{|\, x_{t_j} - x_{t_j-1}\,|^{1/N} : 1 \le j \le p\} \le \epsilon$$

is not satisfied, then go to Step 1. Otherwise take the value

$$z_q^* = \min\{z_i : 1 \le i \le q\}$$

as an estimate of the global minimum and corresponding to this value points $y_q^* \in D$ as estimates of global minimizers and Stop.

The algorithm described above belongs to the class of parallel characteristical methods introduced in [5]. In comparison with the other algorithms from the class [5] the main peculiarity of the new method consists in the following. Instead of usage the global Hőlder constant H or its estimates, PLT executes an adaptive local tuning on the behaviour of the reduced one-dimensional function $f(x)$. The local tuning is based on the estimates μ_j of the local Hőlder constants H_j for every subinterval $[x_{j-1}, x_j], 2 \le j \le q(l)$, during every iteration $l > 1$.

The value μ_j from (3) represents the result of a balance between the global and local information accumulated during the search. If the interval $[x_{j-1}, x_j]$ is small then, the local information (see (4)) represented by λ_j is very important. The global information (see (5)) represented by γ_j is less important because the estimate μ of the global Hőlder constant (see (2)) could be obtained at an interval being very far from $[x_{j-1}, x_j]$. Vice versa, when the interval $[x_{j-1}, x_j]$ is wide, then we cannot trust the local information from (3) and γ_j plays the most important part. The parameter ξ reflects our idea that $f(x)$ is such that over every subinterval $[x_{j-1}, x_j]$ its Hőlder constant $H_j \ge \xi$.

Let us now consider convergence properties of the new method. Let $\{x^q\}$ be the sequence of the trial points generated by PLT during minimizing the function $f(x)$. The first theorem on a level with other results asserts that only local minimizers of $f(x)$ can be limit points of $\{x^q\}$.

Theorem 1. Let $\bar{x} \in [0,1]$ be a limit point of the sequence $\{x^q\}$ and

$$p(l) \le Q < \infty, \ l > 1. \tag{8}$$

Then the following results take place :

i. if $\bar{x} \in (0,1)$ then, there exist two subsequences of $\{x^q\}$ such that the first one converges to $\bar{x}$ from the left and the second one - from the right;

ii. the point $\bar{x}$ is a local minimizer of $f(x)$ if the function $f(x)$ has a finite number of local extrema;

iii. if a part $\bar{x}$ there exists another limit point $\hat{x}$, then $f(\bar{x}) = f(\hat{x})$;

iv. for all $q \geq 1$ it follows $z^q = f(x^q) \geq f(\bar{x})$.

Proof. Since PLT belongs to the class of parallel characteristical methods these results can be easily deduced from the general convergence theory presented in [5]. ∎

The second theorem describes sufficient convergence conditions of the sequence $\{x^q\}$ to a global minimizer x^*. This result can not be obtained within the general framework from [5] and we prove it. In order to proceed let us introduce $\{q\} = \{1,2,3,\ldots\}$ as the sequence enumerating iterations executed by PLT.

Theorem 2. If (8) is true and there exists an infinite subsequence $\{h\}, \{h\} \subseteq \{q\}$, such that for the interval $[x_{j-1}, x_j], j = j(l), l \in \{h\}$, containing the point x^* during the lth PLT iteration the following inequality takes place

$$r\mu_j \geq 2^{1-1/N} K_j + (4^{1-1/N} K_j^2 - M_j^2)^{1/2}, \tag{9}$$

where

$$K_j = \max\{(z_{j-1} - f(x^*))(x^* - x_{j-1})^{-1/N}, (z_j - f(x^*))(x_j - x^*)^{-1/N}\}, \tag{10}$$

$$M_j = |z_{j-1} - z_j| (x_j - x_{j-1})^{-1/N}, \tag{11}$$

then, x^* is a limit point of $\{x^q\}$.

Proof. In order to start the proof we notice that the following inequality takes place for the estimates μ_j from (3):

$$0 < \xi \leq \mu_{j(l)} \leq \max\{H, \xi\} < \infty, \qquad 2 \leq j \leq q(l), l \geq 1. \tag{12}$$

Suppose now that there exists a limit point $x' \neq x^*$ of the trial sequence $\{x^q\}$. From (6), (12) and the first assertion of Theorem 1 we conclude for the interval $[x_{i-1}, x_i], i = i(l)$, containing x' at the lth iteration of PLT, that

$$\lim_{l \to \infty} R(i(l)) = -4f(x'). \tag{13}$$

Consider now the interval $[x_{j-1}, x_j], j = j(l)$,

$$x^* \in [x_{j-1}, x_j] \tag{14}$$

and suppose that x^* is not a limit point of $\{x^q\}$. This means that there exists an iteration number m such that for all $l \geq m$

$$x^{q(l)+k} \notin [x_{j-1}, x_j], \qquad j = j(l), 1 \le k \le p(l+1),$$

i.e. new trial points will not fall in the interval (14). Estimate now the characteristic $R(j(l)), l \ge m$, of this interval. It follows from (10) and (14) that

$$z_{j-1} - f(x^*) \le K_j(x^* - x_{j-1})^{1/N},$$

$$z_j - f(x^*) \le K_j(x_j - x^*)^{1/N}.$$

By summrizing these two inequalities and by using the designation

$$\alpha = (x^* - x_{j-1})/(x_j - x_{j-1}),$$

we obtain

$$z_{j-1} + z_j \le 2f(x^*) + K_j((x^* - x_{j-1})^{1/N} + (x_j - x^*)^{1/N}) =$$

$$= 2f(x^*) + K_j(\alpha^{1/N} + (1-\alpha)^{1/N})(x_j - x_{j-1})^{1/N} \le$$

$$\le 2f(x^*) + K_j(x_j - x_{j-1})^{1/N} max\{\alpha^{1/N} + (1-\alpha)^{1/N} : 0 \le a \le 1\} =$$

$$= 2(f(x^*) + 2^{-1/N}K_j(x_j - x_{j-1})^{1/N}).$$

From this estimate, (9), and (11) we have

$$R(j(l)) = r\mu_j(x_j - x_{j-1})^{1/N} + (z_{j-1} - z_j)^2(r\mu_j)^{-1}(x_j - x_{j-1})^{-1/N} - 2(z_{j-1} + z_j)$$

$$\ge (x_j - x_{j-1})^{1/N}(r\mu_j + M_j^2(r\mu_j)^{-1} - 2^{2-1/N}K_j) - 4f(x^*) \ge -4f(x^*) \quad (15)$$

for all iteration numbers $l \in \{h\}$.

Since x^* is a global minimizer and the sequence $\{h\}$ is infinite, then from (13) and (15) it follows that an iteration number l^* will exist such that

$$R(j(l^*)) \ge R(i(l^*)). \tag{16}$$

But, in according with the decision rules of PLT, this means that during the l^*th iteration one of $p(l^*)$ new trials will be executed at the interval (14). Thus, our assumption that x^* is not a limit point of $\{x^q\}$ is not true and theorem has been proved. ∎

Corollary 1. Given the conditions of Theorem 2 all the limit points of the sequence $\{x^q\}$ are the global minimizers of $f(x)$.

Proof. The corollary follows from the third assertion of Theorem 1. ∎

Let X^* be the set of the global minimizers of the function $f(x)$. Corollary 1 ensures that the set of limit points of the sequence $\{x^q\}$ belongs to X^*. Conditions ensuring coincidence of these sets are established by Corollary 2.

Corollary 2. If condition (14) is fulfilled for all the points $x^* \in X^*$, then the set of limit points of $\{x^q\}$ coincides with X^*.

Proof. Again, the corollary is a straightforward consequence of Theorem 2 and the third assertion of Theorem 1. ∎

Theorem 2 and its corollaries are very important both from theoretical and practical viewpoints. It is known (see [2], [16], [17]) that every point $y \in D$ can have up to 2^N images on the curve. The global minimizer x^* can have up to 2^N images on the curve too. To obtain an ϵ-approximation of x^* it is enough to find only *one* its image on the curve.

We have proved that to have convergence to a global minimizer x^* PLT needs the fulfilment of condition (9) (which is considerably weaker than the Lipschitz condition) for one of the images of the point x^* on the curve. Thus, (in contrast with the other methods from the class [5]) the new parallel method does not need the exact value of the precise Lipschitz constant L (neither its upper estimate) for the *whole* region D. It is enough that condition (9) is fulfilled in a neighbourhood of x^* for *one* its image on the curve. In contrast with this, methods using in their work the exact Lipschitz (Hölder) constant (or its upper estimate) will have convergence to *all* 2^N *images* on the curve. Of course, this fact leads to a significant slowing down the search and explains why PLT works faster. Results of numerecal experiments to be presented in the next Section confirm that the new approach permits to accelerate the search significantly.

3.   Numerical experiments

In this section we compare performance of the new method PLT with the original information algorithm (IA), parallel information algorithm (PIA), and sequential information algorithm with local tuning (IALT). An ALLIANT FX/80 parallel mini-supercomputer having 4 processors has been used on the series of 100 two-dimensional multiextremal functions from [4] usually applied for testing information GO methods (see [4], [5], [12], [16], [18]):

$$f(x) = \sqrt{\left[\sum_{i=1}^{7}\sum_{j=1}^{7}(A_{ij}a_{ij}(x) + B_{ij}b_{ij}(x))\right]^2 + \left[\sum_{i=1}^{7}\sum_{j=1}^{7}(C_{ij}a_{ij}(x) - D_{ij}b_{ij}(x))\right]^2},$$

(17)

where $0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1$ and

$$a_{ij}(x) = \sin(i\pi x_1)\sin(j\pi x_2), \qquad b_{ij}(x) = \cos(i\pi x_1)\cos(j\pi x_2),$$

and $A_{ij}, B_{ij}, C_{ij}, D_{ij}$ are random coefficients from the interval [-1,1].

For all the methods in all the experiments we have used the 12-order approximation of the Peano curve, initial points $\{0.2, 0.4, 0.6, 0.9\}$, the reliability parameter $r = 2.9$ and the search accuracy $\epsilon = 0.001$. We have chosen $\xi = 10^{-6}$ in PLT.

In Tables 1, 2 we present average results for 100 functions from the class (17). The column "%" shows a quantity of experiments in which global minima have been found. In Tab. 1 we compare the sequential method IALT and PLT. It is seen from the table that the introduced type of the parallelizm permits to achive high levels of speed up in comparison with the sequential method IALT. Note, that obtaining speed up higher than the number of processors used in the parallel case is possible due to parallel adaptive estimating the local Hőlder constants.

In order to underline the effect obtained after introducing the local tuning, in Tab. 2 we report the values of speed up obtained by using the parallel method PIA in comparison with the original sequential IA (see [18]). Both methods use an adaptive estimate of the *global* Hőlder constant in their work.

In Tab.3 we compare PLT working with the local Hőlder constants with PIA using the global Hőlder constant. The data from Tabs. 1, 2 have been used. The obtained values of speed up both in time and in trials are shown. It is seen from Tab. 3 that in comparison with PIA the new method PLT functions faster more than 4.5 times in trials and more than 3.5 times in time.

Table 1. Average results of the numerical experiments executed by PLT w.r.t. the sequential method IALT.

Method	Processors	%	Trials	Time	Speed up (trials)	Speed up (time)
IALT	1	98	351.37	11.15	-	-
PLT	2	96	339.00	4.22	1.97	3.33
PLT	3	98	349.75	2.73	3.02	7.15
PLT	4	98	348.24	1.99	3.94	10.33

Table 2. Average results of numerical experiments with the information algorithms using global estimates of the Hőlder constant.

Method	Processors	%	Trials	Time	Speed up (trials)	Speed up (time)
IA	1	100	1575.12	70.04	-	-
PIA	2	100	1596.08	21.06	1.97	3.33
PIA	3	100	1562.61	9.80	3.02	7.15
PIA	4	100	1599.92	6.78	3.94	10.33

Table 3. Speed up obtained by using PLT in comparison with PIA on the data from Tabs. 1, 2.

Processors	Speed up in trials	Speed up in time
2	4.71	4.99
3	4.47	3.59
4	4.59	3.41

4. A brief conclusion

In this paper a new parallel algorithm for solving global optimization multidimensional problems has been proposed. The method unifies two powerful approaches for accelerating the search : parallel computations and local tuning on the behavior of the objective function. Peano-type space-filling curves have been used to reduce the multidimensional problem to the one-dimensional one. For the obtained problem the new method adaptively estimates local Hőlder constants over different subintervals of the one-dimensional search region to tune itself on the local behaviour of the reduced objective function. The local information is used by the method over the *whole* search region *during* the global search in contrast with traditional approaches usually starting a local search in a *neighbourhood* of a global minimizer *after* stopping the global procedure. It has been theoretically and numerically shown that the new technique permits to accelerate the search considerably.

Acknowledgments

References

1. Bertocchi, M. (1990), A parallel algorithm for global optimization, *Optimization*, **62(3)**, 379–386.
2. Butz, A.R. (1968), Space filling curves and mathematical programming, *Inform. Control.*, **12(4)**, 314–330.
3. Gergel, V.P. and Sergeyev, Ya.D. Sequential and parallel global optimization algorithms using derivatives, (to appear in *Computers and Mathematics with Applications*).
4. Grishagin, V.A. (1978), Operation characteristics of some global optimization algorithms, *Problems of Stochastic Search*, **7**, 198–206.
5. Grishagin, V.A., Sergeyev, Ya.D. and Strongin, R.G. (1997), Parallel characteristical global optimization algorithms, *J. of Global Optimization*, **10**, 185–206.
6. Floudas, C.A. and Pardalos, P.M. (1996), Eds., *State of the Art in Global Optimization*, Kluwer Academic Publishers, Dordrecht.
7. Horst, R. and Pardalos, P.M. (1995), Eds., *Handbook of Global Optimization*, Kluwer Academic Publishers, Dordrecht.
8. Migdalas, A., Pardalos, P.M., and Storøy, S. (1997), Eds., *Parallel Computing in Optimization*, Kluwer Academic Publishers, Dordrecht.

9. Pardalos, P.M., Phillips, A.T., and Rosen, J.B. (1992), *Topics in Parallel Computing in Mathematical Programming*, Science Press.

10. Schendel, U. (1984), *Introduction to Numerical Methods for Parallel Computers*, Ellis Horwood, Chichester, England.

11. Sergeyev Ya.D. (1995) A one-dimensional deterministic global minimization algorithm, *Comput. Maths. Math. Phys.*, **35(5)**, 705–717.

12. Sergeyev, Ya.D. (1995), An information global optimization algorithm with local tuning, *SIAM J. Optimization*, **5(4)**, 858–870.

13. Sergeyev, Ya.D. (1998), Global one-dimensional optimization using smooth auxiliary functions, *Mathematical Programming*, **81(1)**, 127–146.

14. Sergeyev, Ya.D., and Grishagin, V.A. (1994), A parallel method for finding the global minimum of univariate functions, *JOTA*, **80**, 513–536.

15. Sergeyev, Ya.D., and Grishagin, V.A. (1994), Sequential and parallel algorithms for global optimization, *Optimization Methods and Software*, vol.3, 111-124.

16. Strongin, R.G. (1978), *Numerical Methods on Multiextremal Problems*, Nauka, Moscow.

17. Strongin, R.G. (1992), Algorithms for multiextremal mathematical programming problems employing the set of joint space-filling curves, *J. of Global Optimization*, **2**, 357–378.

18. Strongin, R.G. and Sergeyev, Ya.D. (1992), Global multidimensional optimization on parallel computer, *Parallel Computing*, **18**, 1259–1273.

19. Sutti, C. (1984), Local and global optimization by parallel algorithms for MIMD systems, *Annals of Operations Research*, **1**, 151–164.

20. Van Laarhoven, P.J.M. (1985), Parallel variable metric algorithm for unconstrained optimization, *Mathematical Programming*, **33**, 68–87.